Efficient implementation of 2D Exner model using GPU

Tomás Morales de Luna

Departamento de Matemáticas
Universidad de Córdoba

In collaboration with

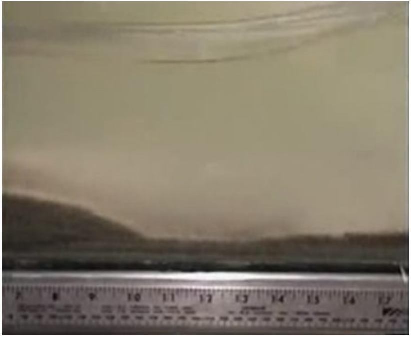
M. de la Asunción, M.J. Castro

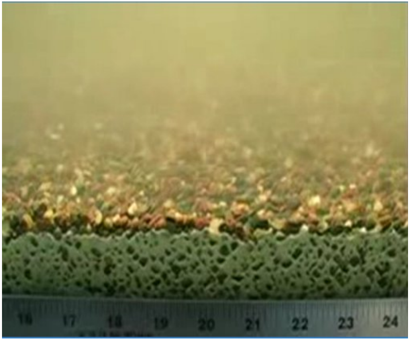
UNIVERSIDAD DE CÓRDOBA

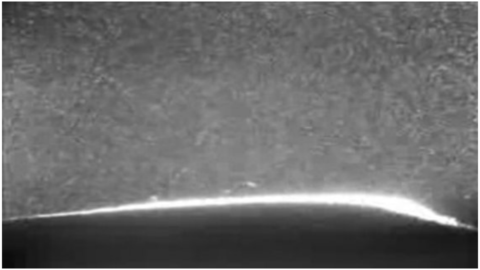
Grupo EDANYA

Outline

Outline

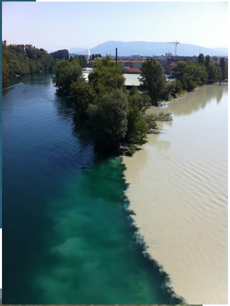



16
17
18
19
20
21
22
23
24

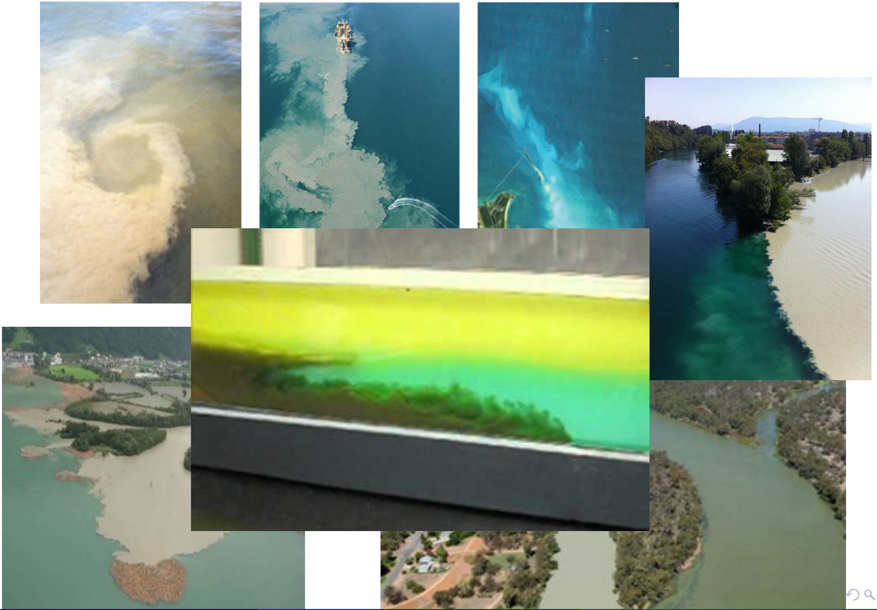


Real flows



Real flows



In this talk...

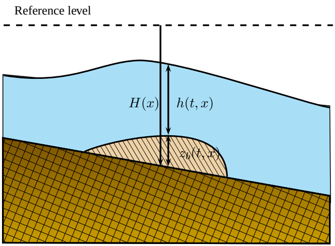
- bedload: [Grass 1981], [M&PM 1948], [Nielsen 1992], [Fowler 2007], [Fraccarollo 05,11], etc.
- suspension and hyperpycnal currents: [Parker 1986], [Kubo 2004], [Khan 2005], [Morales et al 2009], [Castro-Fernández-Morales 2015], etc.

Why sediment modeling is interesting?

- Profound impact on the morphology of continental shelves, lakes, artificial dams, etc...
- Destructive effect on pipelines, cables, submerged structures, ...

Outline

Exner equations



$$\begin{cases} \partial_t h + \partial_x(hu) = 0, \\ \partial_t(hu) + \partial_x(hu^2 + g/2h^2) \\ \qquad\qquad = gh\partial_x(H - z_b) + S_f(W), \\ \partial_t z + \partial_x q_b = 0 \end{cases}$$

- $W = \begin{pmatrix} h & q & z_b \end{pmatrix}^T$
- $S_f(W)$ is the friction term
- $q(t,x) = h(t,x)\,u(t,x)$

Outline

Grass Model

$$q_b(h, hu) = \frac{1}{1-\varphi} A_g u|u|^{m_g - 1}, \quad 1 \le m_g \le 4$$

- A_g constant that depends on grain size, viscosity, etc.
- φ is the porosity
- Very simple model
- Critical shear stress set to zero (movement starts when $|u| > 0$)

Bottom shear stress

$$\tau_b = \rho_w g h S_{\mathrm{f}},$$

Darcy-Weisbach

$$S_{\mathrm{f}} = \frac{f u|u|}{8gh}$$

Manning

$$S_{\mathrm{f}} = \frac{n^2 u|u|}{h^{4/3}}$$

Shield's parameter

$$\theta = \frac{|\tau_b|}{(\rho_s - \rho_{\mathrm{w}})gd_s}$$

Shield's parameter

$$\theta = \frac{|\tau_b|}{(\rho_s - \rho_{\mathrm{w}})g d_s}$$

Bedload transport flux

$$q_b = \frac{Q}{1-\varphi}\Phi(\theta)\,\mathrm{sgn}\,(\tau_b)$$

$$Q = \sqrt{\left(\frac{\rho_{\mathrm{s}}}{\rho_{\mathrm{w}}} - 1\right) g d_{\mathrm{s}}^3}$$

Meyer-Peter&Müller model (1948)

$$q_b(h, hu) = \frac{Q}{1-\varphi} \cdot \Phi(\theta) \cdot \operatorname{sgn}(\tau_b) \qquad \text{(MPM)}$$

$$\Phi(\theta) = 8(\theta - \theta_c)_+^{3/2}$$

- More complex but gives good results
- Sediment transport starts if $\theta > \theta_c$
- Valid for small slopes (lower than 2%)

Fernández Luque & Van Beek (1976)

$$q_b(h, hu) = \frac{Q}{1-\varphi} \cdot \Phi(\theta) \cdot \operatorname{sgn}(\tau_b)$$

$$\Phi(\theta) = 5.7(\theta - \theta_c)_+^{3/2}$$

Ribberink (1998)

$$q_b(h, hu) = \frac{Q}{1-\varphi} \cdot \Phi(\theta) \cdot \operatorname{sgn}(\tau_b)$$

$$\Phi(\theta) = 11(\theta - \theta_c)_+^{1.65}$$

Nielsen (1992)

$$q_b(h, hu) = \frac{Q}{1-\varphi} \cdot \Phi(\theta) \cdot \operatorname{sgn}(\tau_b)$$

$$\Phi(\theta) = 12\sqrt{\theta}(\theta - \theta_c)_+$$

General formulation

$$q_b(h, hu) = \frac{cQ}{1-\varphi}(\theta)^{m_1}(\theta - \theta_c)_+^{m_2}\operatorname{sgn}(\tau_b),$$

Properties

- $\dfrac{\partial q_b}{\partial h} = -k\dfrac{q}{h}\dfrac{\partial q_b}{\partial q}$ with $k = 1$(Darcy-Weisbach) or $k = 7/6$ (Manning)

Properties

- $\dfrac{\partial q_b}{\partial h} = -k\dfrac{q}{h}\dfrac{\partial q_b}{\partial q}$ with $k = 1$(Darcy-Weisbach) or $k = 7/6$ (Manning)
- Grass and models based on Darcy-Weisbach always hyperbolic

Properties

- $\dfrac{\partial q_b}{\partial h} = -k\dfrac{q}{h}\dfrac{\partial q_b}{\partial q}$ with $k = 1$(Darcy-Weisbach) or $k = 7/6$ (Manning)
- Grass and models based on Darcy-Weisbach always hyperbolic
- Manning models at least hyperbolic for $|u| < 6\sqrt{gh}$

Properties

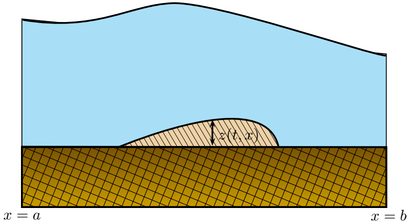
- $\dfrac{\partial q_b}{\partial h} = -k \dfrac{q}{h} \dfrac{\partial q_b}{\partial q}$ with $k = 1$(Darcy-Weisbach) or $k = 7/6$ (Manning)
- Grass and models based on Darcy-Weisbach always hyperbolic
- Manning models at least hyperbolic for $|u| < 6\sqrt{gh}$
- The models are only valid for small slopes, otherwise gravity effects should be included

Common problems to these expressions

- q_b depends only on the variables h and u and does not take into account sediment layer

Common problems to these expressions

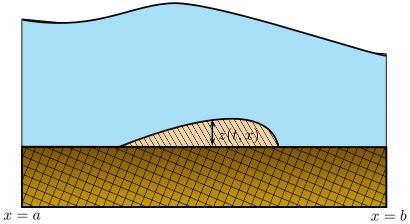
- q_b depends only on the variables h and u and does not take into account sediment layer



$$\partial_t z_b + \xi \partial_x q_b \qquad = 0$$

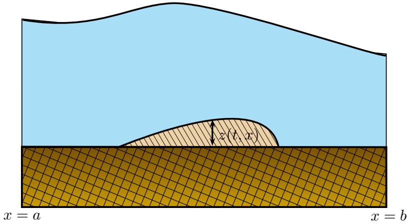
Common problems to these expressions

- q_b depends only on the variables h and u and does not take into account sediment layer



$$\int_0^T \int_a^b \left(\partial_t z_b + \xi \partial_x q_b\right) dxdt = 0$$

- q_b depends only on the variables h and u and does not take into account sediment layer



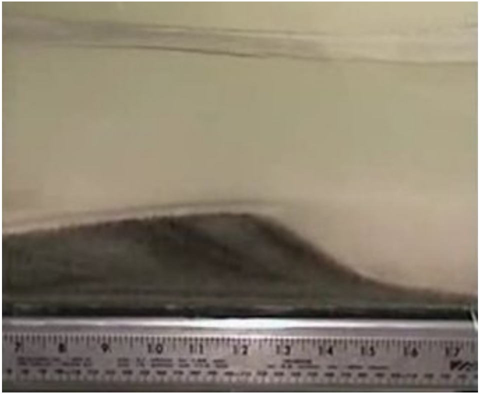
$$\int_a^b z_b|_{t=T}\,dx - \int_a^b z_b|_{t=0}\,dx = -\xi \int_0^T \left(q_b|_{x=b} - q_b|_{x=a}\right) dt$$

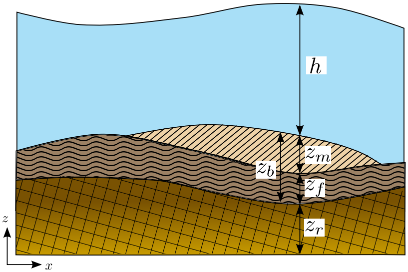
Sediment mass is not preserved!!

Modified models (Fowler type models)

$$q_b(h, hu, z_b) = Q\, c\, \frac{z_b}{z_b^0} \theta^{m_1} (\theta - \theta_{cr})_+^{m_2} \operatorname{sgn}(\tau_b) \qquad \text{(modFow)}$$

where z_b^0 is a characteristic thickness of the sediment layer.





$$\begin{cases} \partial_t h + \partial_x(hu) = 0, \\ \partial_t(hu) + \partial_x(hu^2 + gh^2/2) = -gh\partial_x(z_m + z_f + z_r), \\ \partial_t z_m + \partial_x \widetilde{q}_b = \dot{z}_e - \dot{z}_d, \\ \partial_t z_f = -\dot{z}_e + \dot{z}_d, \end{cases}$$

$$\widetilde{q}_b = z_m v_b,$$

[Fernández-Nieto, Lucas, Morales, Cordier 2014]

Extension to 2D

General formulation

$$\mathbf{q_b}(h, hu, hv) = \frac{cQ}{1-\varphi}(\theta)^{m_1}(\theta-\theta_c)_+^{m_2}\operatorname{sgn}(\boldsymbol{\tau_b}),$$

$$\theta = \frac{\|\boldsymbol{\tau_b}\|}{(\rho_s - \rho_{\mathrm{w}})gd_s}$$

$$\operatorname{sgn}(\boldsymbol{\tau_b}) = \frac{\boldsymbol{\tau_b}}{\|\boldsymbol{\tau_b}\|}$$

Extension to 2D

General formulation

$$\mathbf{q_b}(h, hu, hv) = \frac{cQ}{1-\varphi}(\theta)^{m_1}(\theta-\theta_c)_+^{m_2}\operatorname{sgn}(\boldsymbol{\tau_b}),$$

$$\theta = \frac{\|\boldsymbol{\tau_b}\|}{(\rho_s-\rho_{\mathrm{w}})gd_s}$$

$$\operatorname{sgn}(\boldsymbol{\tau_b}) = \frac{\boldsymbol{\tau_b}}{\|\boldsymbol{\tau_b}\|}$$

$$\frac{\partial \mathbf{q_b}}{\partial h} = -k\frac{q^x}{h}\frac{\partial \mathbf{q_b}}{\partial q^x} - k\frac{q^y}{h}\frac{\partial \mathbf{q_b}}{\partial q^y}$$

Outline

Outline

Numerical Scheme

$$W_i^{n+1} = W_i^n - \frac{\Delta t}{\Delta x}\left(\mathcal{D}^+_{i-1/2}\left(W_{i-1}^n, W_i^n, H_{i-1}, H_i\right) + \mathcal{D}^-_{i+1/2}\left(W_i^n, W_{i+1}^n, H_i, H_{i+1}\right)\right)$$

where

$$\mathcal{D}^{\pm}_{i+1/2}\left(W_i^n, W_{i+1}^n, H_i, H_{i+1}\right) = \frac{1}{2}\Big[\mathcal{F}\left(W_{i+1}^n\right) - \mathcal{F}\left(W_i^n\right) + B_{i+1/2}\left(W_{i+1}^n - W_i^n\right)$$
$$- S_{i+1/2}\left(H_{i+1} - H_i\right) \pm Q_{i+1/2}\left(W_{i+1}^n - W_i^n - \left(A_{i+1/2}\right)^{-1} S_{i+1/2}\left(H_{i+1} - H_i\right)\right)\Big].$$

being

$$\mathcal{F}(W) = \begin{pmatrix} q \\ \dfrac{q^2}{h} + \dfrac{g}{2}h^2 \\ \xi\, q_b \end{pmatrix}, \quad B_{i+1/2} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & gh_{i+1/2} \\ 0 & 0 & 0 \end{pmatrix}, \quad S_{i+1/2} = \begin{pmatrix} 0 \\ gh_{i+1/2} \\ 0 \end{pmatrix},$$

Numerical Scheme

$$A_{i+1/2} = \begin{pmatrix} 0 & 1 & 0 \\ gh_{i+1/2} - u_{i+1/2}^2 & 2u_{i+1/2} & gh_{i+1/2} \\ \left(\dfrac{\partial q_b}{\partial h}\right)_{i+1/2} & \left(\dfrac{\partial q_b}{\partial q}\right)_{i+1/2} & 0 \end{pmatrix}.$$

- The Polynomial Viscosity Matrix $Q_{i+1/2}$ is defined using the PVM-IFCP method [Fernández-Nieto 2011]:

$$Q_{i+1/2} = \alpha_0 Id + \alpha_1 A_{i+1/2} + \alpha_2 \left(A_{i+1/2}\right)^2$$

  where α_0, α_1 and α_2 are defined from the eigenvalues of $A_{i+1/2}$.
- Finally, $S_f(W)$ is discretized in a semi-implicit way.

Numerical Scheme

$$A_{i+1/2} = \begin{pmatrix} 0 & 1 & 0 \\ gh_{i+1/2} - u^2_{i+1/2} & 2u_{i+1/2} & gh_{i+1/2} \\ \left(\dfrac{\partial q_b}{\partial h}\right)_{i+1/2} & \left(\dfrac{\partial q_b}{\partial q}\right)_{i+1/2} & 0 \end{pmatrix}.$$

- The Polynomial Viscosity Matrix $Q_{i+1/2}$ is defined using the PVM-IFCP method [Fernández-Nieto 2011]:

  $$Q_{i+1/2} = \alpha_0 Id + \alpha_1 A_{i+1/2} + \alpha_2 \left(A_{i+1/2}\right)^2$$

  where α_0, α_1 and α_2 are defined from the eigenvalues of $A_{i+1/2}$.
- Finally, $S_f(W)$ is discretized in a semi-implicit way.

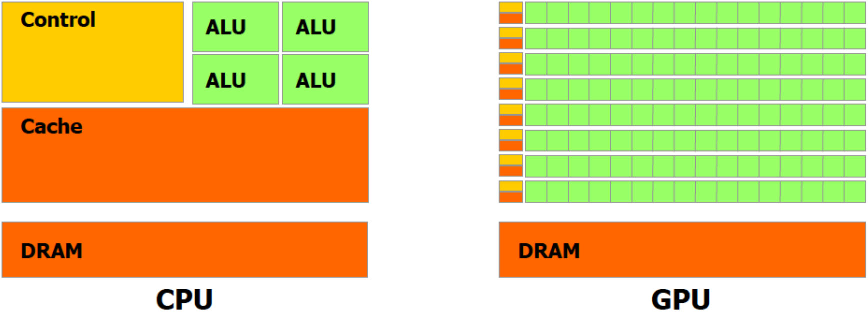
Second order numerical scheme

- TVD Runge-Kutta time discretization.
- MUSCL-type reconstruction operator for triangular meshes.

Outline

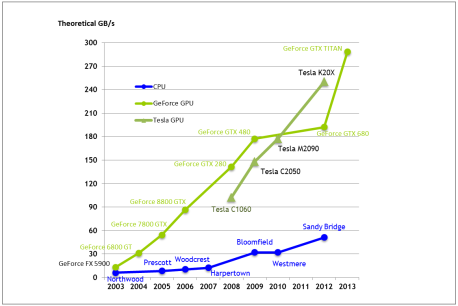
What is a GPU?

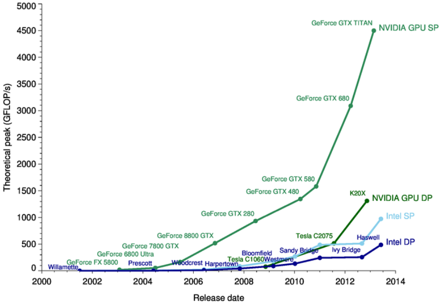
CPU vs GPU architectures



- GPUs are specialized circuits designed to rapidly manipulate the building of images in a a frame buffer intended for output to a display.
- GPUs are specialized for compute-intensive highly parallel computations.
- GPU are especially well-suited to address problems that present a high data-parallel paradigm.
- The same program is executed for each data element in GPUs, so the flow control of the code is simple.

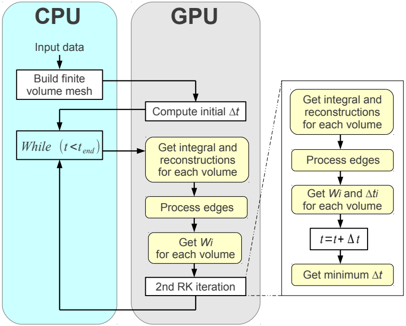
Why GPUs?





- Also price!!!

CUDA Implementation



- Each GPU step is assigned to a CUDA kernel.

- Full code in double precision.
- Edge data (normals and volume indexes) in global memory.
- Volume data in 1D textures.
- Stencil data (volume indexes) in 1D texture.
- In global memory: an auxiliary state, the barycenters of the volumes, 12 vectors for the reconstruction coefficients, and all the reconstructed values.
- The creation of the stencils and the computation of the reconstruction coefficients is performed at the beginning.
- 6 accumulators in global memory to write the edge contributions.

Outline

Outline

Conical Dune

- Test proposed in [Hudson 2001].
- Grass model.
- Mesh size: 64000 triangles.
- Simulation time: 100 hours.
- Incoming $u_x = 1$ m/s
- CFL $= 0.7$
- $\rho_0 = 0.4$
- $A_g = 10^{-3}$
- Spread angle of 22.3° using a mesh of 13755 triangles (asymptotic $\approx 21.79°$)

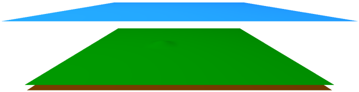
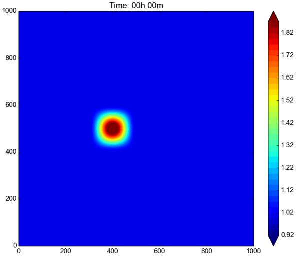
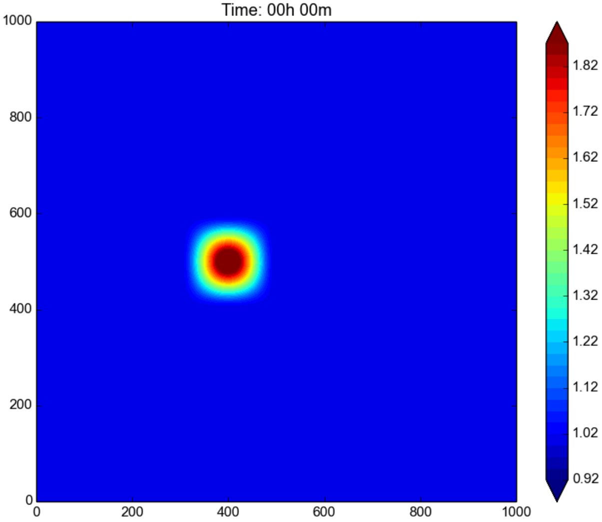


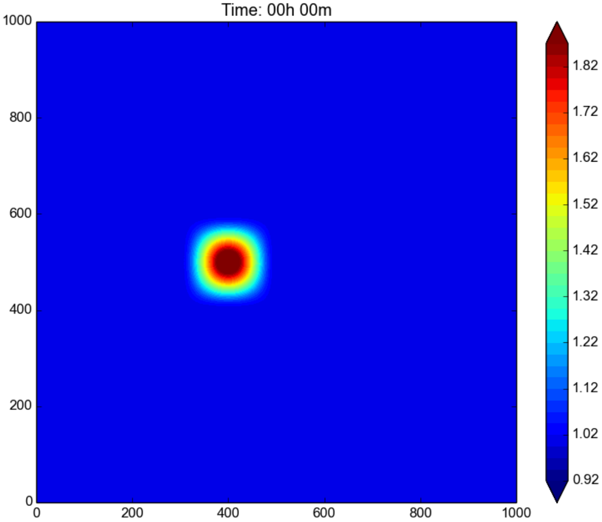
Figure: Initial state



Conical Dune



Conical Dune



Analysis of Efficiency

- GeForce GTX Titan Black.
- 64000 triangles, 100 hours.

Order	Iterations	Runtime	MVols/s
1st	5755088	2h 18m 39s	44.3
2nd	5754913	5h 19m 08s	19.2

Order	Most costly kernel
1st	Edge processing (**90** % of runtime)
2nd	Edge processing (**78** % of runtime)

Outline

L-shaped channel

- MPM model.
- Mesh size: 128002 triangles.
- Simulation time: 1 hour.
- Incoming $u_y = -0.5$ m/s
- CFL $= 0.9$
- $\rho_0 = 0.4$
- $\rho_{\rm w} = 1000$ kg/m^3
- $\rho_{\rm s} = 2600$ kg/m^3
- $d_{\rm s} = 10^{-3}$ m

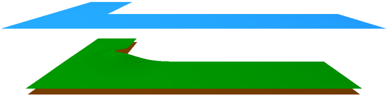
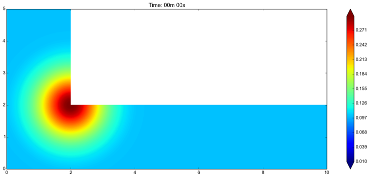
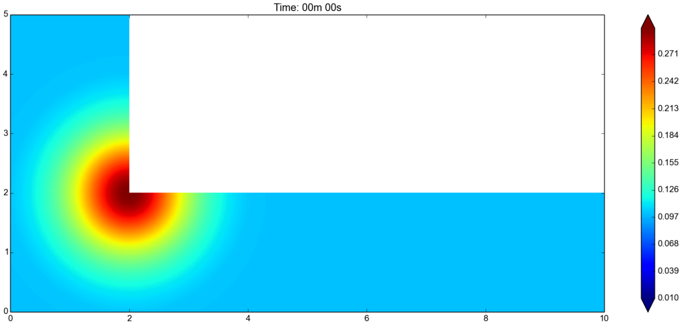


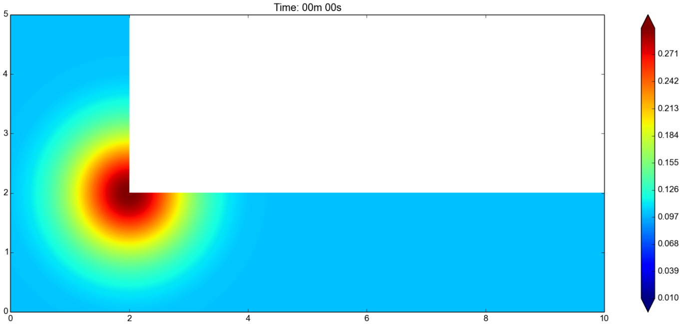
Figure: Initial state



L-shaped channel



L-shaped channel



Analysis of Efficiency

- GeForce GTX Titan Black.
- 128002 triangles, 1 hour.

Order	Iterations	Runtime	MVols/s
1st	4521681	7h 03m 36s	22.8
2nd	4612758	15h 49m 17s	10.4

Order	Most costly kernel
1st	Edge processing (**95** % of runtime)
2nd	Edge processing (**87** % of runtime)

Outline

Scour Around a Spur Dike

- Laboratory experiment [Mioduszewski 2003].
- MPM model.
- Mesh size: 31016 triangles.
- Simulation time: 40 seconds
- Incoming $q_x = 0.005$ m^3/s
- CFL $= 0.5$
- $\rho_0 = 0.4$
- $\rho_s/\rho_w = 1.4$
- $d_s = 1.28$ mm
- Preliminary results adjust well with laboratory results.

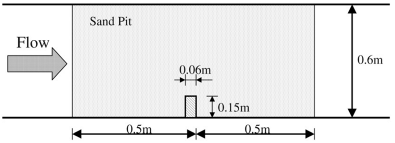
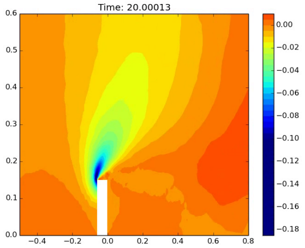
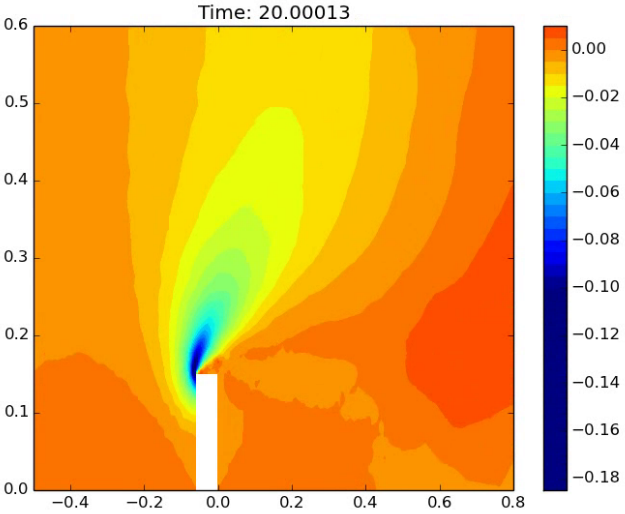


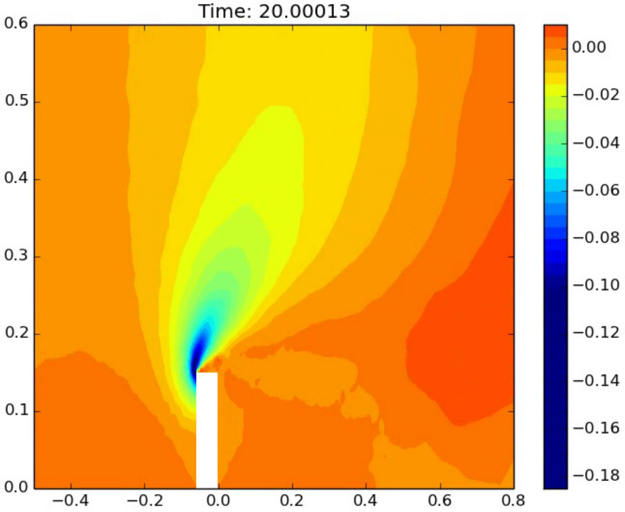
Figure Source: [Liu 2008]



Scour Around a Spur Dike



Scour Around a Spur Dike



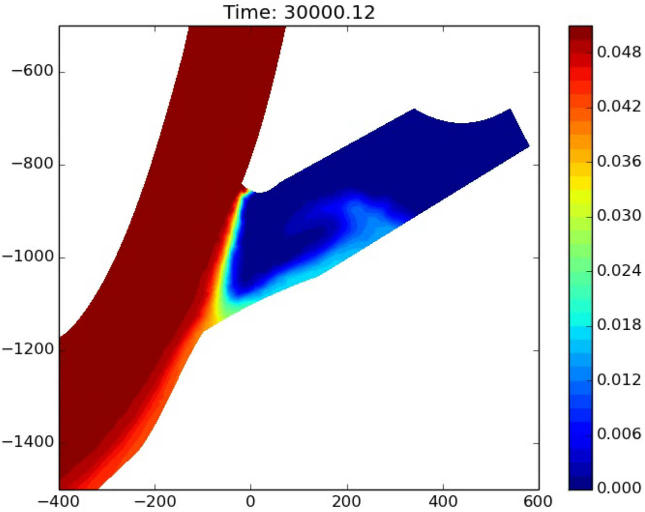
Outline

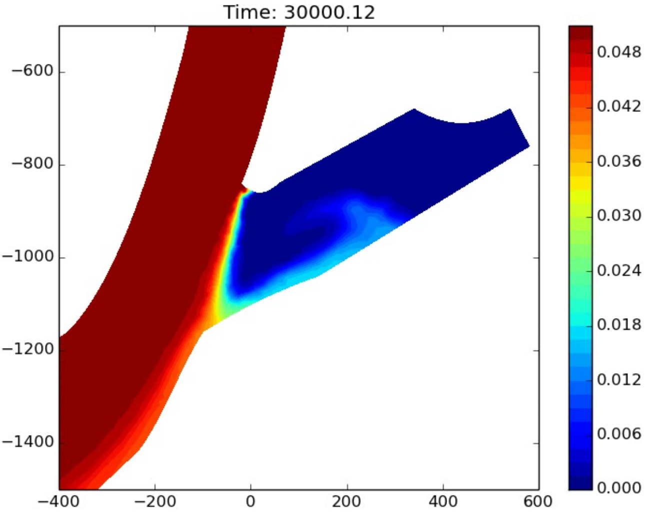
Conclusions

- First and second order GPU implementations of the Grass and MPM bedload sediment transport models using triangular meshes and double numerical precision.
- The numerical scheme is robust and preliminary results adjust well with the angle of evolution of a dune and some laboratory experiments.

Future Work

- Work with real domains and more complex models.
- Hyperpycnal model example (Guadalquivir river):


Time: 30000.12
−600
−800
−1000
−1200
−1400
−400
−200
0
200
400
600
0.048
0.042
0.036
0.030
0.024
0.018
0.012
0.006
0.000


Time: 30000.12
−600
−800
−1000
−1200
−1400
−400
−200
0
200
400
600
0.048
0.042
0.036
0.030
0.024
0.018
0.012
0.006
0.000

Thank you for your attention

Tomas.Morales@uco.es